

IN THE CLAIMS

Claims 1-29 are pending in the application and are presented for reconsideration.

1. (Currently Amended) A computer implemented method for dynamically rendering data in a markup language, the method comprising:

identifying a symbol in the data in the markup language, the symbol indicating a query of a data set;

accessing the data set in order to generate a resolution to the query; and

dynamically rendering the resolution to the query as a part of the markup language, according to at least one rule associated with the markup language.
2. (Original) The method of claim 1, wherein:

the symbol comprises a delimited token.
3. (Original) The method of claim 1 wherein:

the symbol is located within the data in the markup language such that the query is associated with a markup language tag.
4. (Original) The method of claim 3 wherein:

the markup language comprises Hyper Text Markup Language.
5. (Currently amended) The method of claim 3 wherein the step of dynamically rendering the resolution further comprises the step of:

dynamically rendering the resolution of the query according to at least one rule associated with the markup language tag with which the query is associated.
6. (Original) The method of claim 1 wherein:

the data set comprises a set of at least one document in a hierarchically structured format.

7. (Original) The method of claim 6 wherein:
the hierarchically structured format comprises Extensible Markup Language.
8. (Original) The method of claim 7 wherein:
the symbol conforms an Extensible Markup Language standard concerning queries.
9. (Original) The method of claim 1 wherein:
the data set comprises a database.
10. (Currently amended) The method of claim 1 wherein [:] the step of dynamically
rendering is performed by a computer browser.
11. (Original) The method of claim 1 wherein:
rendering is performed by software running on a hand held computing device.
12. (Original) The method of claim 1 further comprising:
generating a resolution to the query by retrieving a node set from a set of documents in
Extensible Markup Language; and
rendering each member of the node set.
13. (Original) The method of claim 1 wherein:
the query contains at least one variable.
14. (Original) The method of claim 13 wherein:
each variable contained in the query comprises a delimited token.
15. (Original) The method of claim 13 wherein:
at least one variable contained in the query is bound to a specific node in hierarchically
structured data.
16. (Original) The method of claim 15 wherein:

the hierarchically structured data comprises a set of at least one document in Extensible Markup Language.

17. (Original) The method of claim 1 wherein rendering the resolution further comprises:

updating the data set.

18. (Original) The method of claim 17 wherein updating the data set further comprises: writing to a set of at least one document in Extensible Markup Language.

19. (Currently amended) A computer program product for dynamically rendering data in a markup language, the computer program product comprising:

program code for identifying a symbol in the data in the markup language, the symbol indicating a query of a data set;

program code accessing the data set in order to generate a resolution to the query;

program code for dynamically rendering the resolution to the query as a part of the markup language, according to at least one rule associated with the markup language; and

a computer readable medium on which the program codes are stored.

20. (Currently amended) The computer program product of claim 19 further comprising: program code for dynamically rendering the resolution of the query according to at least one rule associated with a markup language tag with which the query is associated.

21. (Original) The computer program product of claim 19 further comprising: program code for generating a resolution to the query by retrieving a node set from a set of documents in Extensible Markup Language; and

program code for rendering each member of the node set.

22. (Original) The computer program product of claim 19 further comprising:

program code for updating the data set.

23. (Original) The computer program product of claim 22 wherein the program code for updating the data set further comprises:

program code for writing to a set of at least one document in Extensible Markup Language.

24. (Original) A computer system for dynamically rendering data in a markup language, the computer system comprising:

an identification module, for identifying a symbol in the data in the markup language, the symbol indicating a query of a data set;

a data access module, for accessing the data set in order to generate a resolution to the query, the data access module being coupled to the identification module; and

a rendering module, for dynamically rendering the resolution to the query as a part of the markup language, according to at least one rule associated with the markup language, the rendering module being coupled to the data access module.

25. (Currently amended) The system of claim 24 wherein:

the rendering module is further for dynamically rendering the resolution of the query according to at least one rule associated with a markup language tag with which the query is associated.

26. (Original) The system of claim 24 further comprising:

a resolution generation module, for generating a resolution to the query by retrieving a node set from a set of documents in Extensible Markup Language, the resolution generation module being coupled to the data access module; and the rendering module is further for rendering each member of the node set.

27. (Original) The system of claim 24 further comprising:

an updating module, for updating the data set, the updating module being coupled to the rendering module.

28. (Original) The system of claim 27 wherein:

the updating module is further for updating the data set by writing to a set of at least one document in Extensible Markup Language.

29. (Original) The method of claim 3 wherein:

the markup language comprises Wireless Markup Language.